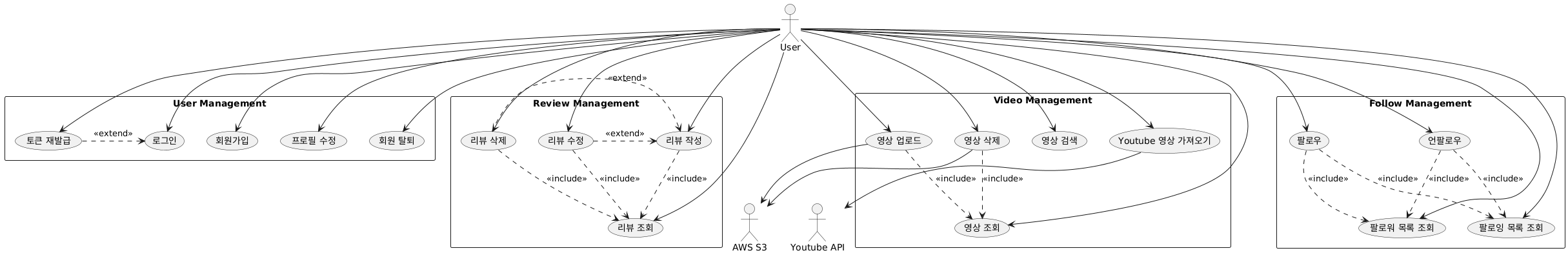
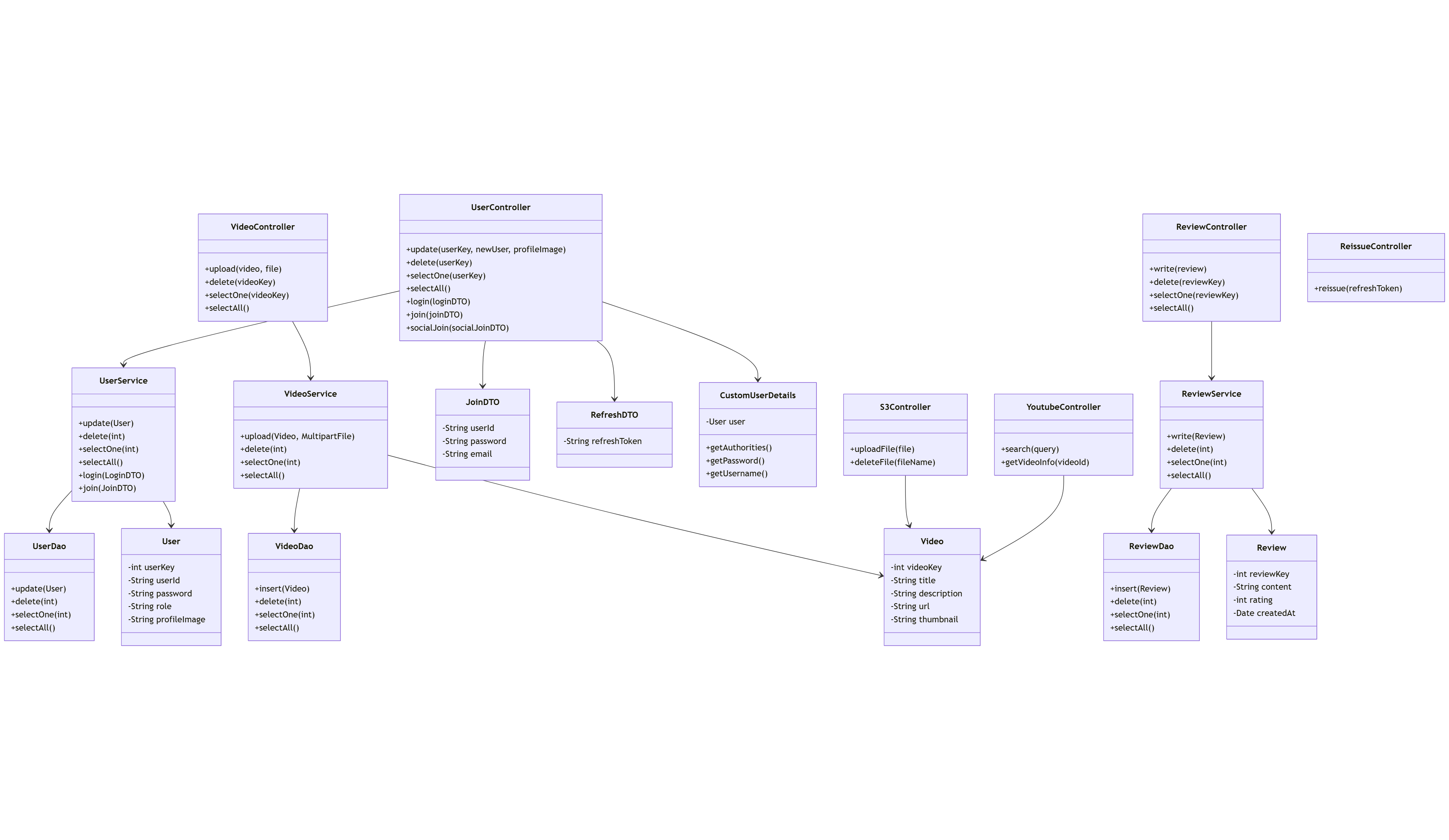
1. 요구사항 정의서

|  |  |  |  |
| --- | --- | --- | --- |
| 대분류 | 기능명 | 요구사항 정의 | 요구사항 명세 |
| User | 사용자 회원가입 | 사용자는 ID, 비밀번호, 이메일을 입력하여 회원가입할 수 있어야 한다. | DTO 양식에 맞춰 값을 받아User 테이블에 해당 정보 삽입 |
| 사용자 로그인 | 사용자는 ID와 비밀번호로 로그인할 수 있어야 한다.  로그인 성공 시 JWT 토큰을 발급한다. | 사용자 ID, PW를 받아 User 테이블에서 ID, PW 함께 일치하는지 확인 후 일치 시에 로그인 성공, 세션 생성 및 저장 / 불일치 시 예외 발생 시 아이디, 비밀번호 확인 View 반환 |
| 사용자 로그아웃 | 로그인한 사용자는 로그아웃 할 수 있어야 한다 | 세션 삭제 |
| 사용자 프로필 이미지 조회 | 사용자의 프로필 이미지 경로를 반환한다. | 사용자 이미지 프로필을 S3 스토리지에 등록 후 String값 반환 |
| 사용자 정보 수정 | 사용자는 프로필 이미지와 함께 자신의 정보를 수정할 수 있어야 한다. | 세션에서 사용자 ID를 받고 DTO 양식에 맞춰 정보 받아 User 테이블에서 해당하는 ID의 사용자 데이터 갱신 |
| 사용자 삭제 | 사용자는 자신의 계정을 삭제할 수 있어야 한다. | 세션에서 사용자 ID 받아 팔로우 테이블에서 해당하는 ID 사용자 삭제. 이후  User 테이블에서 해당하는 ID의 사용자 데이터 삭제 |
| 사용자 단일 조회 (+ 마이페이지) | 특정 사용자의 정보를 조회한다. | 사용자의 정보를 받아 옴. |
| 사용자 검색 | 입력된 문자열로 사용자 ID를 검색한다. | 회원 ID를 받아  User테이블에서 조회, 반환 |
| Video | 영상 업로드 | 사용자는 유튜브 링크와 메타 데이터를 함께 업로드할 수 있어야 한다.  사용자는 영상 파일과 메타데이터를 함께 업로드할 수 있어야 한다. | (유튜브 링크 업로드)  Youtube 영상 링크(쿼리 속 id) 포함,  DTO 양식에 맞춰 DB저장(영상 유형 컬럼 : Y)  (S3 스토리지 업로드)  동영상 파일 업로드 및 서버 DB(DTO양식에 맞춰)와 S3 스토리징에 저장 후 DB에는 S3 ID 저장(영상 유형 컬럼 : S) |
| 영상 수정 | 사용자가 등록한 영상 정보를 수정할 수 있어야 한다. | (유튜브)  ID단일 조회 후  DTO양식에 맞춰 수정 데이터 받기(ID 제외) Video 테이블 수정을 하고, ID를 통해 youtube 테이블 속 youtubeID 수정  (S3 스토리지)  ID단일 조회 후 DTO양식에 맞춰 수정 데이터 받기(ID 제외)  만약file != null이면 ID에 해당하는 S3 경로의 영상 삭제 및 재업로드 후 변경된 경로를 반환하고 변경된 경로 포함,  DTO 양식에 맞춘 정보로 Video 테이블 수정하며 ID를 통해 S3 테이블 속 경로 수정 |
| 영상 삭제 (단건 및 채널 전체) | 사용자가 업로드한 영상을 삭제할 수 있어야 하며, 채널 이름으로 전체 삭제도 가능해야 한다. | (유튜브)  ID단일 조회 후 ID를 통해 youtube 테이블 속 youtubeID 삭제하며 Video 테이블에서 일치하는 ID 삭제  (S3 스토리지)  ID단일 조회 후 ID를 통해 S3테이블 속 경로 반환하여 S3에서 해당 경로의 영상 삭제  그리고 S3테이블에서 데이터 삭제 후 Video 테이블에서 일치하는 ID 튜플 삭제 |
| 영상 조회 (단일 및 리스트) | 특정 영상 하나 또는 사용자가 업로드한 영상 목록을 조회한다. | (유튜브)  영상 id(Video테이블에서 유형 Y 확인 후 유튜브 id 테이블에서 일치하는 값 반환)을 youtube list()로 반환하여 API로 영상 보여주기  영상 id(Video테이블에서 유형 Y 확인 후 유튜브 id 테이블에서 일치하는 값 전체 반환)을 youtube list()로 반환하여 API로 영상 보여주기  (S3 스토리지)  영상 id(Video 테이블에서 유형 D 확인 후 S3테이블에서 일치하는 경로 반환)을 S3로 반환하여 화면으로 해당 영상 보여주기  영상 id(Video 테이블에서 유형 D 확인 후 S3테이블에서 일치하는 경로 반환)을 S3로 반환하여 화면으로 해당 영상 보여주기 |
| 영상 검색 | 영상 검색 키워드를 기반으로 영상 리스트를 반환한다. | 검색 옵션 String을 받아 영상 유형 별 다수 조회 실행 + 검색 조건(앞에서 받은 검색 옵션 문자열) 동적 쿼리로 적용하여 영상 조회 결과 반환 |
| 영상 정렬 검색 | 조회수, 업로드 순 등 다양한 기준으로 정렬된 영상을 검색한다. | 검색한 영상 리스트를 기준에 맞게 정렬하여 반환한다. |
| Review | 리뷰 등록 | 사용자는 영상에 대해 리뷰를 작성할 수 있어야 한다. | 리뷰 등록 |
| 리뷰 수정 | 작성자는 자신의 리뷰를 수정할 수 있어야 한다. | 리뷰 수정 |
| 리뷰 삭제 | 작성자는 자신의 리뷰를 삭제할 수 있어야 한다. | 리뷰 삭제 |
| 리뷰 조회 (단건 및 영상 별 전체) | 특정 리뷰 또는 영상 별 리뷰 리스트를 조회할 수 있어야 한다. | 리뷰 조회 |
| Log | 운동 로그 등록 | 사용자는 자신의 운동 기록을 저장할 수 있어야 한다. | 세션에서 ID + DTO에 맞춰 데이터(날짜, 운동량(시간, 칼로리, 운동 부위) 받아 운동량 테이블에 데이터 생성 |
| 일별 운동 조회 | 특정 날짜의 운동 기록을 조회할 수 있어야 한다. | (월 단위에서 시작) 세션에서 ID + 월 단위에서 일자를 선택하여 운동 로그 테이블에서 ID+운동일자에 해당하는 데이터 반환 받아 DTO 양식에 담아 반환 |
| 월별 운동 조회 | 특정 월의 운동 날짜 목록을 조회할 수 있어야 한다. | 세션에서 ID 받아 운동량 테이블에서 해당 ID의 운동 일자 전부 반환받아 존재하는 일자에 맞게 달력에 표지 |
| 부위별 통계 조회 | 사용자는 운동 부위별 칼로리 소모량 통계를 확인할 수 있어야 한다. | 세션에서 사용자 ID를 받아 운동 로그 테이블에서 운동 부위별로 그룹핑하여 칼로리 총합을 계산 (GROUP BY 부위) 하여 JSON 형태로 각 부위별 칼로리 총합 반환 |
| 전체 운동 로그 조회 | 사용자 전체 운동 로그를 조회할 수 있어야 한다. | 세션에서 사용자 ID + 조회 하려는 사용자 ID 받아 팔로우 테이블에서 팔로잉 유저 팔로우 중인지 확인 + 팔로잉 테이블에서 팔로우 유저 팔로잉 중인지 확인  만약 if true : 조회하려는 사용자 ID의 운동 정보 페이지(월단위)로 이동, 일 선택 시 일단위 운동 정보 페이지로 이동 |
| Follow | 팔로우, 언팔로우 | 사용자는 다른 사용자를 팔로우 및 언팔로우를할 수 있어야 한다. | (단일 사용자 조회가 완료된 상태)세션에서 사용자 ID + 단일 조회가 된 ID 받아 팔로우 테이블에 팔로잉(단일 조회 ID), 팔로워(세션 속 ID) 튜플 삽입 |
| 팔로잉/팔로워 조회 | 사용자 기준으로 팔로잉/팔로워 리스트를 조회할 수 있어야 한다. | 세션에서 사용자 ID 받아 팔로우 테이블\_followingID 가 일치하는 사용자 모두 반환  세션에서 사용자 ID 받아 팔로우 테이블\_followerID 가 일치하는 사용자 모두 반환 |
| 팔로우 여부 확인 | A 사용자가 B 사용자를 팔로우 중인지 여부를 확인할 수 있어야 한다. | 세션에서 사용자 A ID가 확인하려는 사용자 B ID 입력받아 팔로우 테이블에서 (followerUser = A AND followingUser = B) 조건으로 검색하여 존재하면 true, 없으면 false 반환 |
| Login | accessToken 재발급 | refreshToken으로 accessToken을 재발급 받을 수 있어야 한다. | refreshToken을 통해 accessToken 재발급 |

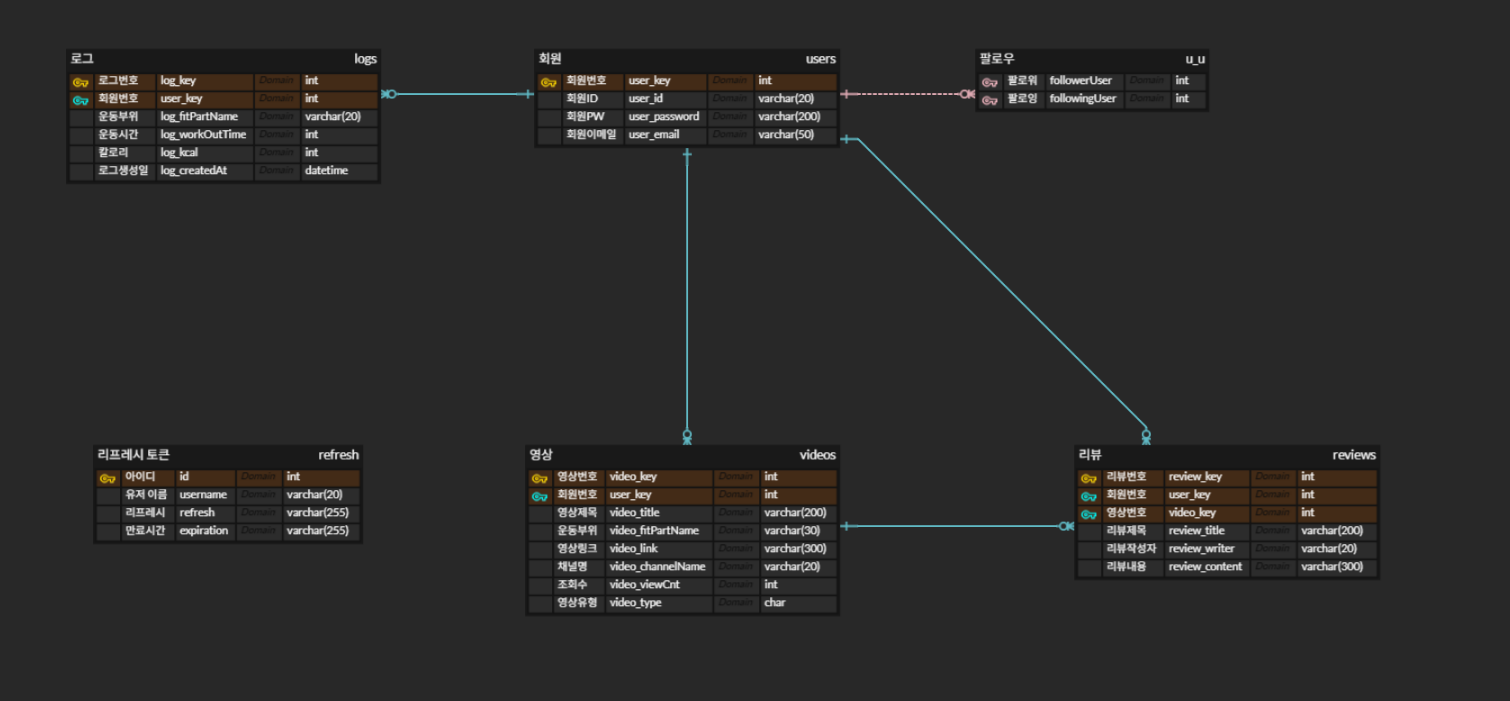
1. Use-Case



1. Class Diagram



1. ER Diagram



1. WBS 차트 + 간트 차트

